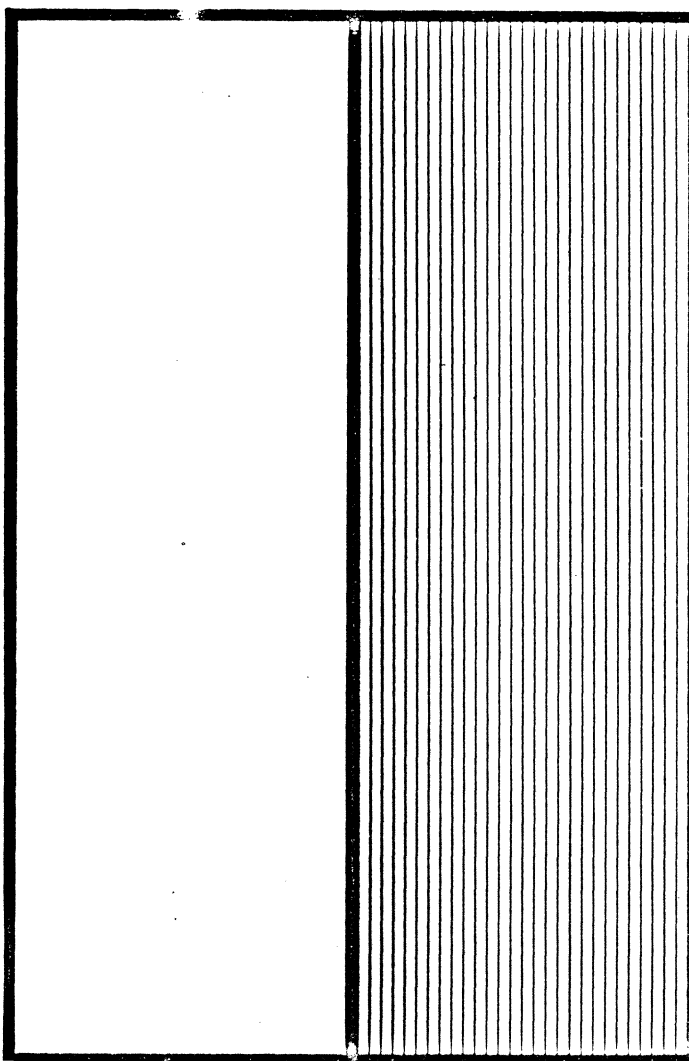


UNIVAC

9400 SYSTEM

PROGRAM DESCRIPTION DRAWING

CUSTOMER
ENGINEERING



SPERRY RAND



UNIVAC

DRAWING NO. 4091702

REVISION

CUSTOMER ENGINEERING
PRODUCT DIAGNOSTIC SOFTWARE

9400 - 5024/8424 or 8425 Disc Handler

SIGNATURE

TITLE

DATE

APPROVAL

APPROVAL

APPROVAL

APPROVAL

REVISION STATISTICS RECORD

4091702

DWG REV	MSTR REV	RELEASE DATE	DESIGNER NAME	EIR NUMBER	FCO NUMBER
-	N		PDS Phila., Pa.	C12607	

REVISION DESCRIPTION RECORD

4091702

REV	DESCRIPTION
-	Release to 9400 Test Library

TABLE OF CONTENTS

<u>SECTION</u>	<u>PAGE</u>
1. INTRODUCTION	6
1.1 Purpose	6
1.2 Major Objectives	6
1.3 Equipment Configurations	6
1.4 Associated Software	6
1.5 Reference Documents	6
2. FUNCTIONAL DESCRIPTION	7
2.1 General Organization	7
2.2 Performing I-O Functions	7
2.3 Message Handling	7
2.4 Entries to DSKH	7
3. OPERATING PROCEDURES	8
3.1 Initialization	8
3.2 Test Program Parameters	8
3.2.1 Action Designators	8
3.2.2 Equipment Designators	8
3.2.3 Program Designators	10
3.3 Test Program Messages	11
4. SUPPLEMENTARY DATA	14
4.1 Supplementary Software Information	14
4.1.1 Parameter Locations	14
4.1.2 Device Addresses	14
4.1.2.1 BAL 12,GETDA	14
4.1.2.2 BAL 12,GETDAP	14
4.1.2.3 BAL 12,GETDA2	14
4.1.3 Issuing I-O Commands	15
4.1.4 Timing Mode	18
4.1.5 Message Handling	19
4.1.6 Test Table	21
4.1.7 Typing Parameters	21
4.2 Supplementary Hardware Information	25
4.2.1 Disc Pack Layout	25
4.2.2 Recording on the Disc	25
4.2.3 Command Codes	28
4.2.4 Additional Information	29
4.2.4.1 Seek Commands	29
4.2.4.2 Search Commands	29
4.2.4.3 Write Commands	30
4.2.4.4 Set File Mask Commands	30
4.2.4.5 Space Count Commands	30
4.2.4.6 Erase Command	31
4.2.4.7 Sense Command	31

TABLE OF FIGURES

<u>FIGURE</u>		<u>PAGE</u>
1.	MCR-CCB	24
2.	Disc Pack Organization	26
3.	Record Formats	27
4.	Sense Data	32
5.	Additional Sense Bytes	33

TABLE

1.	DSKH Entries	19
2.	DSKH Tags	20

1. INTRODUCTION

1.1 Purpose. This document describes a 9400 Maintenance test program for the 5024/8424 of 25 Subsystem.

1.2 Major Objectives. The objective of this test program is to perform a test of the 5024/8424 or 25 Subsystem. Test organization and operator options are intended to satisfy the needs of Design, Quality Control and Field Engineering personnel in checkout, acceptance and maintenance activities.

1.3 Equipment Configurations. All equipment configurations must be connected to a Selector I-O channel of the 9400 Processor. This test will operate the following equipment configuration:

- Disc File Control - Type 5024
- Disc Drive - Type 8424 or 25
- Disc Pack Assembly - Type F12 14-01

1.4 Associated Software. This program has DSKH included as an integral part of the program. This is the standard disc handler routine used by all Disc Maintenance test programs. It is the interface between the program and MCR, which performs program load, parameterization, execution, deletion, I-O handling, and message handling.

1.5 Reference Documents

DRAWING

DESCRIPTION

S70040

UNIVAC 9400 Processor, Type 3019-00 and -01.

--

UNIVAC 9400 Dual I-O Selector Channel, Type
XXXX-00 and -01.

P10076

Disc File Control, Type 5024.
8424 or 25

4091622

Maintenance Control Routine.

4091623

Parameter and Message Routines.

4091670

5024/8414 Disc Program Handler.

2. FUNCTIONAL DESCRIPTION

- 2.1 General Organization. Every program that utilizes DSKH has the initial entry point of the program go directly to DSKH. DSKH requests, accepts, and processes all parameter typeins. Parameters useful to the operation of individual tests are supplied to those tests in certain tagged locations. Other parameter typeins are used by DSKH to make up tables of device addresses, tests to be entered, etc. As each test is entered, it applies to DSKH for a device address. After a test has been completed, it applies for another address. When it receives an end sentinel instead of an address, it is finished, and it returns to DSKH, which then finds the next test to be entered.
- 2.2 Performing I-O Functions. Every I-O function to be performed by the program requires that a table in DSKH be set up correctly and DSKH be entered at a specific point. Variations of the entry point will automatically preset various table entries that must be entered in the program in a specific format. Certain special entries exist to perform the following commands or command chains:
- Search, Read or Write
 - Seek, Read HA, Search, Read or Write
 - Recalibrate
 - Test I-O
 - Sense
- 2.3 Message Handling. An entry exists in DSKH for printing error messages, and another entry prints other messages.
- 2.4 Entries to DSKH. All entry points to DSKH referred to in the above paragraphs are described in more detail in paragraph 4.

3. OPERATING PROCEDURES

409170

The disc handler routine (DSKH) is used by all disc programs as a standard interface between the programs and the Maintenance Control Routine (MCR) and the Parameter and Message Routine (PMR). Thus the following information holds for all the disc test programs.

3.1 Initialization. Pretest setup consists of the following:

- Mount discs on those handlers to be utilized. These discs may or may not have data on them, depending on which program is to be run. Except for the "Prep" program 02070 all programs expect Home Address (HA) and Track Descriptor Record (RO) on every track of the disc.
- Load the MCR.
- Load the program by typing in its name. All source code programs have a 1 appended to their names for the object programs.

3.2 Test Program Parameters. Since most parameters are used exclusively by DSKH, they will all be described here. Any slight variations for particular programs will be described under that program. All programs are entered and initiated thru the handler, which immediately types out the message:

ENTER 8414 DISC PARAMETERS

It will then loop thru MCR until appropriate parameters have been entered. Two types of parameters will be accepted. They are Action Designators and Control Designators. For conformity, the Control Designators may further be divided into Equipment Designators and Program Designators.

3.2.1 Action Designators. As the name implies, Action Designators specify some action that the Parameter Analysis portion of the program must take. The following Action Designators are recognized by this program:

<u>Parameter</u>	<u>Meaning</u>
------------------	----------------

A	Use the following Control Parameters to replace those in the parameter tables.
D	Eliminate the following Control parameters from those already in the parameter tables.
B	Start the program.
E	Suspend the program. Do not reenter it until a B or an R Action Designator has been typed in.
R	Return to the program at its point of suspension.
V	View the parameters by console PMR-out

3.2.2 Equipment Designators. These define the equipment which is to be operated. Three such parameters are recognized by the program:

<u>Parameter</u>	<u>Meaning</u>
------------------	----------------

Cn	This parameter sets the channel number(s) for the disc handler. n = 1 or 2 for the appropriate selector channel number.
----	--

When two control units are in use, two different channel numbers will be accepted in the form Cn, n.

3.2.2 EQUIPMENT DESIGNATORS: (CONTINUED)

4091702

409170

//Js d - This parameter sets the sub-channel and device numbers for the disc handler.

J = arbitrary EBCDIC alphabetic character to signify that the device is a disc pack.

s = 0 thru F for the appropriate sub-channel number.

To enter two sub-channels, the following parameter format is necessary:

//Js₁d₁ : //Js₂d₂

3.2.2 Equipment Designators (continued)

4091702

Parameter Meaning

continued . . .

d = This is the device address typein. Up to 8 devices may be connected to a 5024, with addresses 0 thru 7. They may be called serially with comma separations, or by a hyphenated typein. If two control units are in use, the devices on the second control unit typed in must be considered as a continuation from the devices on the first control unit. They will then have addresses 8 thru 15.

Fn Three features may exist in a disc system. They are as follows:

F1 = Scan Feature
 F2 = Record Overflow Feature
 F3 = Two Channel Feature

3.2.3 Program Designators. The remaining control designators are for program operation. They are as follows:Parameter Meaning

Tn Up to 31 tests may be incorporated into one program. Each may be called separately, a consecutive string may be called by using a hyphen between the first and last of the string, or a combination of the two may be used. Each individual program will have its own T parameters defined.

Vn Certain other variables may be entered by typing a variable number or a variable number followed by data.

Variables preselected in DSKH include V1, V4, V13 and V20/1.

V1 = Allow error printouts.

V3/n = Attempt to recover from errors up to M₁₆ times.

V4 = Stop on error message typeouts. Do not return to program until a reply is typed in.

V6/aaaa = One or two hexadecimal data patterns may be typed in, each being two bytes long.

V7 = Direct all messages to the printer except error and operator action messages.

V8/t = Stop at end of test number t. If t is 0, stop at end of current test. This will set the Suspend Action Designator after typing out an end test message. A Resume Action Designator must be typed in to continue testing.

V11 = Direct ALL messages to the printer. **** check message on printer **** will be typed on the console if operator action is required.

3.2.3 Program Designators (continued)Parameter Meaning

- V13 = Deletes all messages except error messages or those requiring operator action.
- V14/n = Perform test n only and loop on it continuously. This disregards any entries under parameter T and suppresses the test number printout after the first entry. Tests all have decimal numbers.
- V15/cc/hh = A starting cylinder and head number, in decimal, for a disc test area.
- V16/cc/hh = An ending cylinder and head number, in decimal, for a disc test area.
- V17/xxx --- x = Up to eight bytes of hexadecimal data for free use by individual programs. Data must be entered, in groups of four hexadecimal bytes and a slash must separate each group of four. Less than four bytes can not be entered.
- V20/x = Run each program x times. x is in decimal format.

Rn All messages that require typed replies will expect this typein, following Action Designator A. Replies are numbered 1 through 6 as follows: These are the only acceptable replies to an error message.

n Meaning

- 1 Repeat subtest causing error.
- 2 Skip subtest causing error and continue program.
- 3 Restart this test.
- 4 Skip this test; start next one.
- 5 Restart this program from the beginning.
- 6 End the program.

3.3 Test Program Messages. Messages within DSKH have to do with program operation and I-O error conditions. Each test has its own error messages, mostly concerned with data errors or special I-O conditions.

4.3.1 On initial entry to the program, the following message will be typed out: ENTER 84 DISC PARAMETERS

4.3.2 If a reply typein of R6 is encountered, the following message will be typed out:

END DISC TEST

3.3 Test Program Messages (continued)

- 4.3.5 When DSKH reads Sense data, if the status is other than channel end and device end, the following message will be typed out:

SENSE ERROR DS ss

- 4.3.6 After any I-O function has been performed, DSKH compares various entries in the I-O table to results received from MCR. If differences exist, the following message is printed:

```

      ERROR    DXXXX-X   DA XXXX
DS XX      SNS XXXXXXXX CCW XXXXXXXXXXXXXXXXXX
XP XX      XXXXXXXX   LAST SEEK CC XX HH XX

```

Where: D gives the disc test name and current subtest number.
 DA is the current device address.
 DS is the device status taken from IC\$SF.
 SNS is sense data read either by MCR or DSKH.
 CCW gives the last CCW that was performed.
 XP is data taken from the I-O table.

3.3.1 Test Program Message Definitions

1) D - Disc Test

The first four numbers identify the disc test. The last number is the current subtest number.

2) DA - Device Address

The device address of the current unit being tested.

3) DS - Device Status

One byte of information showing the received status of the control unit and device.

The bit information for DS is described in Section 5.0.

4) SNS - Control Unit Sense Information

Sense data acquired from the control unit.

5) CCW - Channel Command Word

These eight bytes contain the command which the equipment was performing when the error occurred.

Byte 0 - 5024 Command
 Bytes 1-3 - The address of the data being transferred.
 Byte 4 - If '40' - command is chained.
 Bytes 5-7 - The length of the data being transferred.

3.3.1 Test Program Message Definitions (continued)

4091702

6) XP - Expected Data

This data is obtained from the I-O table and represents the expected device status and control unit sense data.

7) LAST SEEK CC HH

Represents the last cylinder and head to which a seek command was issued. If non-existent, a seek command has not been performed.

4. SUPPLEMENTARY DATA

4.1 Supplementary Software Information. Most of the following information is required by programmers using DSKM for their own programs.

4.1.1 Parameter Locations. After parameters have been typed in and approved, they will be used, where necessary, to set certain conditions within the handler. Parameters that are required by the program will be located in certain permanent parameter locations. The following table describes these permanent locations. The entries shown under FORMAT show the initial conditions of these parameters, except for the last three. They are initially set to 00000000 and 00CA0013 and variable 17 is all zeros.

<u>NAME</u>	<u>FORMAT</u>	<u>MEANING</u>	
PFEAT	00	Bit 1 = SCAN FEATURE	(F1)
		Bit 2 = RECORD OVERFLOW FEATURE	(F2)
		Bit 3 = 2 CHANNEL FEATURE	(F3)
PVBLE	4C00	Bit 1 = Allow error printouts	(V1)
		Bit 4 = Stop on errors	(V4)
		Bit 8 (PVBLE+1, Bit 0) = Stop after test	(V8)
PRPEAT	05	REPEAT 5 TIMES ON ERRORS	(V3)
PTTERN	55AAAA55	TWO 2-BYTE DATA PATTERNS	(V6)
PSDTA	00CC00HH	START DISC TEST AREA	(V15)
PEDTA	00CC00HH	END DISC TEST AREA	(V16)
PV17	xxxxxxxx xxxxxxxx	Free Variable for use by programs	(V17)

4.1.2 Device Addresses. On initial entry, go to Handler to obtain a channel and device address.

4.1.2.1 BAL 12,GETDA will supply one address, two bytes long, in location DA1. Each succeeding entry will give the next device in the list. After the last device has been supplied, the next entry will store FFFF in DA1. Another entry will restart the list.

4.1.2.2 BAL 12,GETDAP will supply two addresses in locations DA1 and DA2. On each succeeding entry, the address in DA2 will be shifted to DA1 and the next address in the list will be placed in DA2. The last entry will supply the last device selected and the first device selected. The next entry will supply FFFF in DA1 and DA2. Another entry will restart the list.

4.1.2.3 BAL 12,GETDA2 will supply a device address for the first channel in DA1, and a device address for the second channel in DA2. If the same number of devices are in use on each channel, then after the last two have been used, FFFF will be supplied in both DA1 and DA2. The two lists will then be restarted. If the lists are of different lengths then FFFF will be supplied only for the list that has reached its end. The other address will have been changed to the next one in that list. The next entry will retain the valid address while restarting the list that has expired.

4091702
4091702

4.1.3 Issuing I-O Command. The ICR requires a Command Control Block (CCB) of 40 bytes for any I-O function to be performed. The handler contains a CCB for the disc programs but it requires some information for the block from the program.

4.1.3.1 The program must contain all its own Channel Command Words (CCW's), except as noted under 4.1.3.5 thru 4.1.3.9. The program must also contain two consecutive address constants specifying the starting and ending locations of each CCW chain that is to be issued. More information than this is required, but since the handler maintains its own I-O table (IOTBLE), once the information has been supplied to the handler, it need not be supplied again until a change is required. The handler has various entries for supplying the other information. Normally the first entry from each selectable subtest will supply all required information. Succeeding entries will supply the CCW addresses and only that other information required if changes are to be made.

4.1.3.2 The entry points to the handler to set up the IOTBLE and the CCB are as follows:

<u>TAG</u>	<u>SETS</u>
REISSU	NOTHING
ISSUE	CCW addresses.
ISSUE 1	CCW addresses, status expected, device address.
ISSUE 2	CCW addresses, status expected, device address, code byte.
ISSUE 3	CCW addresses, status expected, device address, code byte, error return.
ISSUE 4	CCW addresses, status expected, device address, code byte, error return, sense data.

The handler table may be accessed directly to change specific entries without making all succeeding changes. The table is as follows:

<u>BYTES</u>	<u>NAME</u>	<u>CONTENTS</u>
0-3	IOTBLE	CCW
4-7		CCW + n*8
8-9	IOTBDA	CCDD
10	IOTBDS	DS
11	IOTBCD	CODE
12-15	IOTBEE	ERROR RETURN ADDRESS
16-21	IOTBSD	SENSE DATA EXPECTED

where CCW = starting address of CCW chain.
 CCW + n*8 = ending address of CCW chain.
 CCDD = channel and device address.
 DS = expected status from CCW chain.
 CODE is a special code byte which directs the handler as follows:

<u>BIT</u>	<u>MEANING</u>
0	Read Sense data after completion of CCW chain. Compare contents of IOTBSD.
1	Do not reissue commands in case of errors.
2	Do not print the standard error message.
3	Do not use the error exit IOTBEE.
4-7	Not assigned.

4.1.3.3 The handler contains two separate CCB's. The main one is named XCCB and is used for all program I-O functions. The other one is called XCCES and is used only by the subroutines TIO and SENSE. When SENSE is performed, it stores the sense data in the appropriate location in XCCB, not in its own CCB. The CCB addresses that may be of use to the user are as follows:

NAME	BYTES	USE
XCCB+IC\$CCW	4	The handler supplies the starting address of the CCW chain to the MCR.
XCCB+IC\$CDN	2	The handler supplies the channel device to the MCR.
XCCB+IC\$SF	2	Status is stored. If Unit Check is set, the MCR automatically reads Sense data and stores the Sense status in the second byte.
XCCB+IC\$BCW	4	MCR stores first 4 bytes of CSW. This contains last command of chain in first byte and final CCW address in low order bytes.
XCCB+IC\$RBC	2	Any remaining byte count from last performed CCW.
XCCB+IC\$SB	6	Sense data storage.
XCCB+IC\$EMN	1	Condition code from last SIO performed, or FF if MCR timed out (no interrupt in 2 seconds).
XCCB+IC\$SIO	4	Time from millisecond counter when SIO was performed.
XCCB+IC\$ST1	4	Time from millisecond counter when interrupt occurred.
XCCB+IC\$ST2	4	Time from millisecond counter when second interrupt occurred (if two interrupts occur).

4.1.3.4 Example of coding to issue a chain of three CCW's.

```

      LA      11,LT
      BAL     12,ISSUE
      .
      .
      .
LT      DC     A(CCW1)
        DC     A(CCW1+3*8)
CCW1    CCW    X'31'SEARCH,X'40',5
        CCW    X'03',CCW1,0,1
        CCW    X'06',AREA,0,8
  
```

4.1.3.5 A special entry to the handler will perform a string of three CCW's as specified by the program. The first CCW is a Search, followed of course, by a Transfer in Channel (TIC). The third CCW can be a Read or a Write. Any Search can be specified. Any Read can be specified. Any Write can be specified except Write Home Address or Write Record Zero. A location called SEARCH must be preset with the Search data. The jump to this entry must be followed by a four byte constant giving actual command code of the type of Search required, a one byte length count for the Search data, actual command code for the Read or Write command, and a one byte data length count for that command. Data for a Write should be placed in a location called AREA and data which is read can be found in this AREA. The name of this subroutine is STRW. Before entering STRW, the entire IOTBLE must have been preset either by a BAL to ISSUE4, or by the program.

To perform a Search HA, Read RO the following would be performed:

1. Move four bytes of HA info into SEARCH.
2. Perform the following:

```
BAL 12,STRW
DC X'39041610'
```

3. Find 16 bytes of RO in location AREA.

The four byte constant following the BAL is defined as follows:

- 39 = function code for Search HA
- 04 = number of bytes for Search
- 16 = function code for Read RO
- 10 = number of bytes to be read (assume 8 bytes of ID and 8 bytes of data)

Areas involved and their lengths are as follows:

```
SEARCH = 16 bytes
AREA   = 256 bytes
SEEK   = 6 bytes*
HA     = 5 bytes*
```

*These areas are described under 4.1.3.6.

4.1.3.6 A special entry will perform exactly the commands described under 4.1.3.5 except that the CCW chain is preceded by two other CCW's. These are a Seek (07), and a Read Home Address. The channel and head numbers required for the Seek must be placed in location SEEK in the form 000000000HH. The first three bytes will always contain zeros, so they need not be altered. The five byte Home Address will be found in location HA. The name of this subroutine is SSTRW.

To perform a Seek Cylinder 4B, Search Key Equal to KEY 15, Read Data 200 bytes, the following would be performed:

1. Move 3 bytes (4B0000) into SEEK+3, move 6 bytes (KEY 15) into SEARCH.
2. Perform the following:

```
BAL 12,SSTRW
DC X'A90606C8'
```

3. Find a five byte Home Address in HA and a 200 byte data record in AREA.

4.1.3.7 A special entry will perform a Recalibrate command. Perform the following: 409,702

BAL 12,RECAL

Control will be returned to the instruction following the BAL. As noted above, the IOTBLE must have been set before entering RECAL.

4.1.3.8 A special entry will perform a Test I-O command. This is always performed by the Handler before any I-O function is performed. It uses a special CCB called XCCBS. Results are not examined by the subroutine. If a busy condition is set, the MCR will not return until a Control Unit End interrupt has occurred. At that time, the CU End will be merged with Busy in XCCBS+ICSSF. Perform the following:

BAL 10,TIO

Control will be returned to the instruction following the BAL. The IOTBLE must have the device address in IOTIDA before entering TIO.

4.1.3.9 A special entry will perform a Sense command. This is done by the Handler if IOTBCD contains a 1 in the Bit 0 position, after the normal interrupt has occurred. It utilizes the special CCB called XCCBS, but the data is stored in XCCB+ICSSB. If status from the normal I-O chain contains a unit check, the MCR always performs a Sense command and stores the data in XCCB+ICSSB. In that case, the Handler does not read Sense data. The program can read Sense data by performing the following:

BAL 10,SENSE

Control will be returned to the instruction following the BAL. The IOTBLE must have the device address in IOTIDA before entering SENSE.

4.1.4 Timing Mode. Since timing mode will not be used in more than two of the programs, (Timing Test and possible Dual Channel Test), it will not be supplied by the handler. However, the following example may aid in understanding how to use it.

	LA 0,TMR	RETURN ADDRESS TO RO
	TMCALL	TIMING MODE CALL
TMR	CLI 7,1	RETURN AND INTERRUPT ENTRY
	BC 8,NORM	NORMAL INTERRUPT ENTRY
	BC 4,INIT	INITIAL ENTRY TO TIMING MODE
	BC 15,ERR	4 SECS EXPIRED OR CONSOLE INTERRUPT

When I-O commands are issued, the MCR will intercept the interrupts, determine their cause, and re-enter the program at TMR. Problem register 0 (loc. 0007) will contain the following:

0 = initial entry to timing mode
 1 = I-O interrupt response
 2 = Console interrupt intervened
 3 = 4 second limit reached

To return to normal operations after Timing Mode, the following may be performed:

LA 0,RTMR
 TMRSE

RTMR continue program or EOI

4.1.5 Message Handling. Messages will be issued by the handler, but it is up to the program to set up its own messages. Paragraph 3. of the 9100 Parameter and Message Routine (PMR) PDD describes how messages are to be made up. Every message must follow the rules of that document. In order to issue a print or type command, load register 0 with the starting address of the message, and do a BAL 12,PMRNT, or BAL 12,TYPE. If the message requires a reply that reply will be found as bit positions in XPRPLY on return to the next line in the program.

4.1.5.1 Every message must be preceded by 4 characters:

BYTE	0	1	2	3
CONTENTS	0	{C} {P}	{*} {^}	{A} {I} {B}

where 0 means output
C = console
P = printer
* = a reply is requested
^ = no reply
A = Operator action required
I = Information
D = Operator decision required

The four characters are followed by a string of pseudo instructions and/or local data commands, as described in paragraphs 3.3.2 and 3.3.3 of the PMR PDD. The character "S" (X'E2') must follow the message.

4.1.5.2 Every message to the console or the printer is preceded by a PMR supplied header. This contains the two least significant bytes of the four byte message header, the time and the program number. The actual message starts in print position 14, and extends thru position 71. A carriage return starts the next line at print position 1.

4.1.5.3 A special message header can be used for the printer only, to delete the PMR header and print up to 132 characters per line. Data is supplied exactly the same way as described in the PMR manual. A carriage return within the message will terminate the line, and succeeding data will start the next print line. The message header must be:

OPAG

The purpose of this special option is to allow printing graphic data which requires the full printer page.

4.1.5.4 If character 2 contains an asterisk then the print routine will not return to the program until a reply has been received. Six different replies will be accepted following an add action designator. They must be in the form:

A~RnS

The reply will set corresponding bits in location XPRPLY, and Rn is described as follows:

REPLY	BIT	MEANING
R1	1	Repeat the command that caused the timeout.
R2	2	Skip the command that caused the timeout and continue on with the program.
R3	3	Restart this portion of the test. The handler will use the address in IOTBEE to return to the program.
R4	4	Skip the rest of this test. Go to the next test.
R5	5	Restart the entire program.
R6	6	End the program.

The message portion of the handler will examine all typed in messages and take the appropriate action for messages R1, R5, and R6. The 1-0 portion of the handler will take the appropriate action for the first three replies if it has issued the error timeout. If the user program has issued an error timeout, it should look at the first three bits of location XPRPLY to determine what action to take.

4.1.5.5 When errors are encountered, a BAL should be performed to the error routine using register 9. When TYPE is entered, register 9 is compared to what it was last time TYPE was entered. If it is different, the new register 9 contents are saved, and an error counter XERCNT is set to 1. Otherwise XERCNT is incremented. If this counter reaches the repeat count, the message will be printed and it will require a reply. Until the repeat count is reached, the program will check the print option. If no print has been requested, the program will be re-entered. If printing has been requested, it will check the stop on errors option. If this option has been selected, it will force a reply to the message. Once the message has been typed, the message routine will determine whether to return to the program or to wait for a type-in.

The replies R1 and R2 should be handled carefully. A command chain which produces an error may require a predecessor chain, or it may be a predecessor to future chains. The return points to the program should take these factors into account.

4.1.5.6 When the Handler encounters an error, it will follow the same rules as described above in determining whether a reply is necessary, and whether to reissue the chain, use the error return, or just return to the program.

4.1.5.7 An entry to use the on-line printer is a subset of the error message routine. The message must start with

OP~I or OP~G

Enter the print routine with BAL 12,PRINT. This will allow any information to be printed, regardless of the error message options. Control will return to the program immediately after the print is performed.

4.1.5.8 All messages start with information supplied by the PPR. The actual message starts in print position 14. (See para. 4.1.5.3 for special print option).

4.1.6 Test Table. The program must set up a table of entry addresses for the specific tests. The first entry must contain a count of the number of tests in the program. Succeeding entries will contain a one byte test number, followed by a three byte address, as follows:

```
TESTS    DC    A(n)
          DC    X'1'
          DC    AL3(TEST 1)
          DC    X'2'
          DC    AL3(TEST 2)
          ;
          DC    X'n'
          DC    AL3(TEST n)
```

This table is part of DSKII and must be placed within the Register 15 range. These addresses are entered into Register 14 for entry to the specific tests.

4.1.7 Typing Parameters. Every type-in must start with an Action parameter. If anything follows the Action type-in, a space must be left before the next character is typed. The type-in must end with an end of message character (S). When more than one Action parameter is to be typed in, they must be separated by a period. Another Action parameter must immediately follow the period. Each of the various Control parameters is separated by a space. A typical type-in might look like this:

A^C1, #JB2-5^T2,4,7-11,18^V1,3/5.B(S)

where:

A = Action designator to set the following parameters.

C1 = Selector channel 1

#JB2-5 = Subchannel B, disc drives 2,3,4 & 5.

T2,4,7-11,18 = Perform tests 2, 4, 7 thru 11, and 18.

V1,3/5 = Set variable 1 (allow error printouts) and set variable 3 to 5 (allow error recovery up to 5 times per error).

. Period is end of sentence, but more to follow.

B= Action designator to begin program.

S= End of message.

4.1.7.1 After an error message has been typed out, a reply might look like this:

D^T9-11.A^R4(S)

which would mean Delete tests 9 thru 11 from the list. Do not try to finish this test, but go directly to the next one in the list.

4.1.8 For the convenience of user programs, all tags within the handler that are of no interest to the programmer, start with the letter X. The tags that are of interest are shown here in two lists. The first list shows entry points and what they perform. The second list shows constants and/or areas, their sizes and function.

Table 1

GO	Initial entry tag to go on END card.
BC 15, RETURN	After completion of any test, to allow handler to find next test to be performed.
BAL 12, ISSUE	To set IOTIME with starting and ending CCW addresses, issue the I-O command for that chain, and check all interrupt conditions.
BAL 12, ISSUE1	Same as ISSUE but also set IOTBDA and IOTBES.
BAL 12, ISSUE2	Same as ISSUE1 but also set IOTECD.
BAL 12, ISSUE3	Same as ISSUE2 but also set IOTBEE.
BAL 12, ISSUE4	Same as ISSUE3 but also set IOTESD.
BAL 12, REISSU	Do not alter the table. Use the same CCW chain.
BAL 10, TIO	Perform a Test I-O command. Results are in a CCB called XCCBS and may be found by using the standard displacement tags (ICSxxx).
BAL 10, SENSE	Perform a Sense command to read six sense bytes into location NCCB-ICSSB.
BAL 12, GETDA	Get next device address in DA1.
BAL 12, GETDAP	Get next two device addresses in DA1 and DA2.
BAL 12, GETDA2	Get next device address on each of two channels, in DA1 and DA2.
BAL 12, STRW	Followed by 4 byte constant to perform Search, TIC, Read or Write. Search data must be in SEARCH, I-O data in AREA.
BAL 12, SSTRW	Same as STRW, but chain is preceded by Seek, Read HA. Seek data must be in SEEK and Home Address will be found in HA.
BAL 12, RECAL	Perform a Recalibrate command and return to the next line of the program.
BAL 12, TYPE	Type all error messages. Assumes R9 used to enter all error routines, or to count times thru a given loop. Tests whether typeouts were requested and whether to stop on errors. Enter by loading R0 with address of prescribed format error message.
BAL 12, PRINT	Entry to either print or type any standard message.

TAG	BYTES	USE
IOTBLE	8	} See paragraph 4.1.3.2.
IOTEDA	2	
IOTBBS	1	
IOTBCD	1	
IOTBEE	4	
IOTBSD	6	
DA1	2	Next device address stored here by entry to GETDA.
DA2	2	Second device address stored here by alternate entries to GETDA.
SEARCH	16	Search data must be stored here before entry to STRW or SSTRW.
SEEK	6	Seek data must be stored here before entry to SSTRW.
AREA	256	Place data here to be written or find read data here on entries to STRW or SSTRW.
HA	5	Find Home Address here after entry to SSTRW.
PPEAT	1	} See paragraph 4.1.1
PVBLE	1	
PRPEAT	1	
PTTERN	4	
PSETA	4	
PETA	4	} See paragraph 4.1.1
TESTS	4(n+1)	
XFRPLY	1	
PV17	8	Free hexadecimal variable.

MCR-CCP

00	IC\$REC	
04	IC\$CCW	
08	IC\$BCW	
0C	IC\$SIO	
10		IC\$EMN
14	IC\$SF	
18	IC\$ST1	
1C	IC\$ST2	
20	IC\$SB	
24		IC\$CDN

REC = Remaining byte count

CCW = Address of first CCW

BCW = Bytes 0 to 3 of CSW

SIO = Time of start I-O

EMN = Condition code or FF if timed out

SF = Status, SF+1 = status of sense if U.C. in SF

ST1 = Time of first interrupt

ST2 = Time of second interrupt (if applicable)

SB = Sense data

CDN = Channel-Device address

FIGURE 1

4.2 Supplementary Hardware Information. The following information can all be found in various documentation referenced in paragraph 1.5. It is included here for the convenience of those using this program and its associated programs.

4.2.1 Disc Pack Layout. The F1224 Disc Pack assembly is made up of eleven separate discs mounted on one spindle. The top surface of the top most disc and the bottom surface of the lowest disc are not used. This leaves twenty intermediate surfaces on which data is recorded. Each surface has its own read/write head, and all heads are attached to a common positioning arm. Thus for any one given arm position, twenty different tracks of the disc pack may be accessed. These twenty tracks are referred to as a cylinder. There are 406 different arm positions, or 406 cylinders of twenty tracks each, for a total of 8120 tracks. The cylinders are numbered 0 thru 405 and the heads are numbered 0 thru 19. The three high order cylinders (403, 404, and 405) are reserved as alternate tracks. If any other track on the disc pack cannot be read or written on successfully, it is flagged as a bad track, and one of the high order tracks is flagged as an alternate, or replacement, for the bad track.

4.2.2 Recording on the Disc. A common Index Marker marks the starting point of every track on the disc pack. The first item written on every track is a "Home Address" (HA), which contains a flag byte, the cylinder number (CC) and the head number (HH). This is followed by a Track Descriptor Record known as Record 0 or R0. The Track Descriptor Record is similar to any other record that may be written on a track with the exception that all other records are preceded by a hardware recognized Address Marker. R0 does not have an Address Marker.

4.2.2.1 Records are divided into two or three separate segments or fields. The first field of any record is called either the ID or the Count (C). The last field is called Data (D). There may or may not be a center field called Key (K). All Count fields are fixed at 8 bytes, plus three bytes generated by and utilized by the hardware. By convention, R0 does not contain a Key field, and it contains an 8 byte data field. All information for all three segments must be supplied by the software. The sixth byte of the Count field gives the Key length. This can be zero to 255 bytes. The seventh and eighth bytes of the Count field give the Data length. This can be 0 to 8191, but the maximum Data length allowable on a track is 7294.

4.2.2.2 I-O commands from the computer must supply the combined byte count of all segments of a record being accessed. If the byte count in the I-O command is greater than that specified in the Count field, the Count field byte count will be utilized. If the command byte count is less than that required the following conditions will occur:

--A read command will be truncated to the length specified in the command.

--All other commands will zero fill any unsupplied bytes.

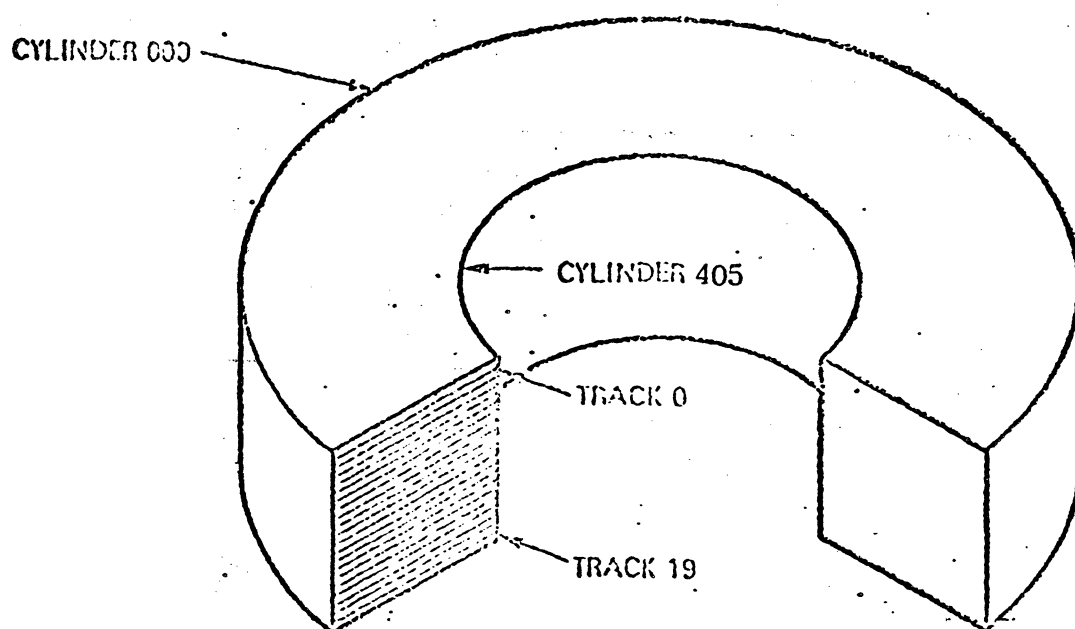


Fig. 2. Gross Data Organization for 8424

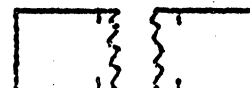
HOME ADDRESS FORMAT

FLAG	CYLINDER	HEAD
BYTE	NUMBER	NUMBER

FLAG BYTE IS SUPPLIED WITH READ OR WRITE.
IT TELLS CONDITION OR STATUS OF THE TRACK.

TRACK DESCRIPTOR OR DATA RECORD FORMAT

FLAG	CYLINDER	HEAD	REC.	KEY	DATA
BYTE	NUMBER	NUMBER	NO.	LGTH	LENGTH



KEY



DATA

FLAG BYTE IS NOT AVAILABLE. IT IS GENERATED AND UTILIZED BY THE HARDWARE.

THE HOME ADDRESS AND EACH SEGMENT OF EVERY DATA RECORD HAS TWO CODE CHECK BYTES APPENDED TO IT BY THE HARDWARE AND UTILIZED BY THE HARDWARE. THEY ARE NOT AVAILABLE.

FIGURE 3

4091702

4.2.3 Command Codes. The Command Code is contained within a Channel Command-Word (CCW) which is 8 bytes in length. A list of consecutive CCW's can be chained together by placing a 1 in bit position 33. This tells the channel to access and operate on the next CCW as soon as the current one has been completed. If bit 33 is a zero, the channel will present ending status as soon as the CCW has been completed.

Bit	0	7	15	31	33	48	63
	COMMAND CODE		DATA ADDRESS		C	BYTE COUNT	

The following table summarizes all the commands executable by the subsystems. The M bit, when zero defines normal operation. When one, operation is in the multiple track mode. The M bit on Initial Program Load is ignored.

COMMANDS	BINARY	HEXADECIMAL
Seeks		
Seek	0000 0111	07
Seek Cylinder	0000 1011	0B
Seek Head	0001 1011	1B
Recalibrate	0001 0011	13
Searches		
Search HA Equal	M011 1001	39 B9
Search ID Equal	M011 0001	31 B1
Search ID High	M101 0001	51 D1
Search ID Equal or High	M111 0001	71 F1
Search K Equal	M010 1001	29 A9
Search K High	M100 1001	49 C9
Search K Equal or High	M110 1001	69 E9
Search KD Equal	M010 1101	2D AD
Search KD High	M100 1101	4D CD
Search KD Equal or High	M110 1101	6D ED
Reads		
Read HA	M001 1010	1A 9A
Read C	M001 0010	12 92
Read RO	M001 0110	16 96
Read D	M000 0110	06 86
Read KD	M000 1110	0E 8E
Read CKD	M001 1110	1E 9E
Initial Program Load	M000 0010	02 82
Writes		
Write HA	0001 1001	19
Write RO	0001 0101	15
Write D	0000 0101	05
Write KD	0000 1101	0D
Write CKD	0001 1101	1D
Write Special CKD	0000 0001	01
Miscellaneous		
Test I-O	0000 0000	00
No Operation	0000 0011	03
Sense	0000 0100	04
Sense Reserve	1111 0100	F4
Sense Release	1101 0100	D4
Set File Mask	0001 1111	1F
Space Count	0000 1111	0F
Erase	0001 0001	11

4.2.3 Command Codes (continued)

One more command exists which is performed by the channel. It is a Transfer In Channel (TIC) command. This instructs the channel to go to the address contained in the command to get the next CCW to be performed by the subsystem. It does not require a chain bit. It has the following format:

0	7	15	31	63
00001000	0---0	ADDRESS	0-----	0

4.2.4 Additional Information

4.2.4.1 Seek Commands. The program must supply a six byte address for each Seek command. Only two of the bytes may contain information. The others must contain zeros.

00	00	00	CC	00	HH
----	----	----	----	----	----

The Seek Head Command (1B) ignores the cylinder entry in CC. Recalibrate does not require any data.

4.2.4.2 Search Commands. Data to be searched for must be supplied by the program, as follows:

00	CC	00	HH
----	----	----	----

Search IIA

00	CC	00	HH	Rn
----	----	----	----	----

Search ID

Search Key or Search KD require varying lengths, depending on what data is being searched for.

4.2.4.3 Write Commands. There are two types of Write commands. When a record exists on a disc its Count field describes the Key and Data lengths. Either the Data portion or the Key and Data portion may be rewritten without affecting anything else on the track. To change a Count field, Count, Key and Data must be rewritten. When this is done it is possible that a new record could overlap some succeeding record previously written. To prevent this, any time a Write CKD is performed, the rest of the track is automatically erased. Of course, successive Write CKD commands, chained together, will be performed with only the gap between records being erased. After the last such command, erasing will be resumed until the index mark is reached. This is called a Format Write. Two other Format Writes exist. They are Write HA and Write RO. Either or both of these commands must be preceded, in the same chain, by one Set File Mask command.

4.2.4.4 Set File Mask Commands. The Set File Mask (SFM) command causes one byte of data to be transferred to the control unit. Only one such command is allowed in any chain, and the mask that is set is only valid for that chain. Whenever a chain is initiated, the existing mask condition is reset to its initial condition of all zeros. The mask allows or restricts various Write commands and/or Seek commands, as follows:

<u>MASK BYTE</u>	<u>WRITES PERMITTED</u>
00000000	All except HA and RO
01000000	None
10000000	Data or Key Data
11000000	All
<u>MASK BYTE</u>	<u>SEEKS PERMITTED</u>
00000000	All
00001000	Head and Cylinder (OB,1B)
00010000	Head (1B)
00011000	None

Any combination of Write Mask and Seek Mask may be combined in one mask byte, but all unused bits must be zeros. It should then have the following form:

ww0ss000

4.2.4.5 Space Count Command. This must supply the three bytes of Key and Data length that are normally picked up by the control unit from the count field. It is used when there is difficulty in reading a count field. The three bytes supplied must be:

KL	DL	DL
----	----	----

When a Space Count command is the first command in a chain, it will search for the index mark, then space over the Home Address. In this case four bytes of HA must be supplied as follows:

00	CC	00	HH
----	----	----	----

4.2.4.6 Erase Command. An erase will not start until an Address Mark is detected, and will terminate as soon as the Index Mark is detected. Thus an Erase after the last Address Mark on a track has been passed will have no effect on the track data.

4.2.4.7 Sense Commands. In normal operation, six bytes of Sense data are read. These six bytes tell everything necessary for program usage. It is possible, however, to read 32 bytes of Sense information. The additional 26 bytes of information may be useful to the engineer in determining the status and various flip-flop settings within the control unit.

4.2.4.7.1 Sense Reserve and Sense Release commands are used only with the Dual Access Control Unit Feature. A Sense Reserve command reserves use of the subsystem for the exclusive use of the channel involved until a Sense Release command has been received. In the meantime the alternate channel will receive a Busy Status in response to commands. At least one byte of Sense data must be read with either of these commands.

4.2.4.7.2 The following figures show the contents of the Sense data.

SENSE DATA
UNIVAC 8411 DISC FILE SUBSYSTEM

BIT	0	1	2	3	4	5	6	7
BYTE 0	COMMAND REJECT	INTERVEN- TION REQ'D	BUS OUT PARITY	EQUIPMENT CHECK	DATA CHECK	OVERRUN	DEFECT OR ALT TRACK	SEEK CHECK
BYTE 1	DATA CHK IN COUNT	TRACK OVERRUN	END OF CYLINDER	INVALID SEQUENCE	NO RECD FOUND	FILE PROTECTED	MISSING ADDR. MARK	OVERFLOW INCOMPLETE
BYTE 2	UNSAFE	Ø	CLOCK CHECK	Ø	Ø	Ø	Ø	Ø
BYTE 3	READY	ON-LINE	UNSAFE	Ø	1	END OF CYLINDER	Ø	SEEK INCOMPLETE
BYTE 4	0	0	0	0	0	0	0	0
BYTE 5	ZEROS EXCEPT ON OVERFLOW INCOMPLETE							
STATUS	ATTENTION	STATUS MODIFIER	CONTROL UNIT END	BUSY	CHANNEL END	DEVICE END	UNIT CHECK	UNIT EXCEPTION

FIGURE 4

ADDITIONAL SENSE BYTES

Byte	Contents
6	Command Register
7	Head Register
8	Track Orientation Register
9	Unit Register
10	Gated Attention Encode
11	Data Length High Register
12	Data Length Low Register
13	Key Length Register
14	Flag Register
15	Mask Register
16	Cyclic Parity High Register
17	Cyclic Parity Low Register
18	File Tag and Binary
19	Binary Counter Low
20	Cylinder Address Register
21	Status Register
22	Queueing Register Stage 3
23	Adder Sum
24	Sequence Counter and Program Counter
25	Miscellaneous Flip-Flops
26	Queueing Register Stage 1
27	Queueing Register Stage 2
28	File Bus Register
29	Data Register
30	X or Y Seek Register
31	Reserved

FIGURE 5